AMS Assignment 4

Xiao Changrong

Oct. 8, 2019

```
# Assignment 4
# Exercise 3
Sigma = matrix(c(1, 0.2, 0.4, -0.5, 0.2, 2, 0.8, 0, 0.4, 0.8, 2, 0, -0.
5, 0, 0, 1), 4, 4)
Rho = cov2cor(Sigma)

get_pca_table = function(pca, Matrix, size) {
    print(summary(pca, loadings = T))
    E = eigen(Matrix)
    print("The eigenvalues: ")
    print(E$values)

    rho = matrix(data = NA, size, size)
    for (i in 1:size) {
        for (j in 1:size) {
            rho[i, j] = E$vectors[i, j] * sqrt(E$values[j]) / sqrt(Matr
ix[i, i])
        }
    }
    print("The correlation coefficients between PCs and variables: ")
    print(rho)
}

pca_cov = princomp(covmat = Sigma)
get_pca_table(pca_cov, Sigma, 4)
```

```
## Importance of components:
##                           Comp.1 Comp.2 Comp.3  Comp.4
## Standard deviation     1.7036 1.2125 1.0870 0.66773
## Proportion of Variance 0.4837 0.2450 0.1969 0.07431
## Cumulative Proportion  0.4837 0.7288 0.9257 1.00000
##
## Loadings:
##      Comp.1 Comp.2 Comp.3 Comp.4
## [1,] -0.234  0.632  0.122  0.729
## [2,] -0.672 -0.376  0.638
## [3,] -0.700        -0.683 -0.190
## [4,]        -0.672 -0.335  0.658
## [1] "The eigenvalues: "
## [1] 2.9023 1.4703 1.1816 0.4459
## [1] "The correlation coefficients between PCs and variables: "
##         [,1]     [,2]    [,3]      [,4]
```

```
## [1,] -0.3986  0.76594  0.1324  0.486759
## [2,] -0.8098 -0.32259  0.4901  0.001778
## [3,] -0.8429  0.07821 -0.5248 -0.089503
## [4,]  0.1048 -0.81439 -0.3646  0.439206
```

```
pca_cor = princomp(covmat = Rho)
get_pca_table(pca_cor, Rho, 4)
```

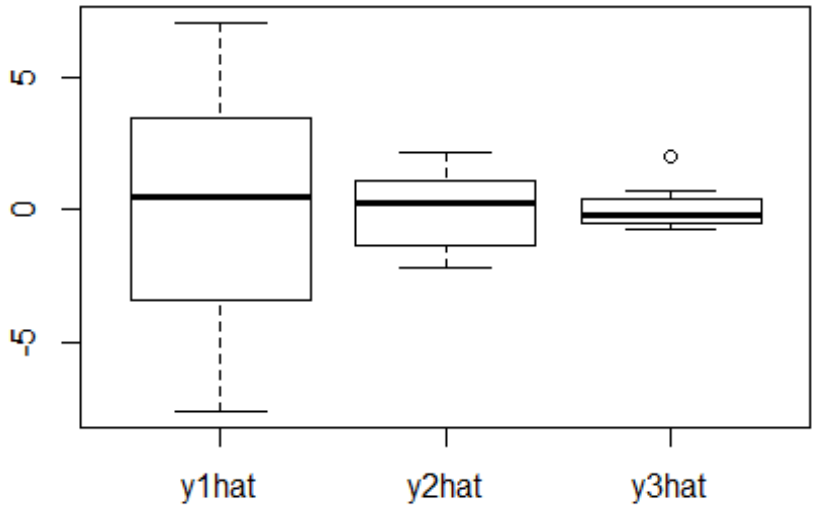
```
## Importance of components:
##                         Comp.1 Comp.2 Comp.3 Comp.4
## Standard deviation     1.2988 1.1260 0.7903 0.6486
## Proportion of Variance 0.4217 0.3169 0.1561 0.1052
## Cumulative Proportion  0.4217 0.7387 0.8948 1.0000
##
## Loadings:
##      Comp.1 Comp.2 Comp.3 Comp.4
## [1,] -0.616 -0.322  0.227  0.682
## [2,] -0.416  0.550 -0.714  0.121
## [3,] -0.496  0.482  0.590 -0.417
## [4,]  0.449  0.601  0.302  0.589
## [1] "The eigenvalues: "
## [1] 1.6869 1.2678 0.6246 0.4207
## [1] "The correlation coefficients between PCs and variables: "
##         [,1]    [,2]    [,3]     [,4]
## [1,] -0.8006 -0.3623  0.1792  0.44236
## [2,] -0.5399  0.6197 -0.5642  0.07866
## [3,] -0.6441  0.5430  0.4661 -0.27031
## [4,]  0.5828  0.6765  0.2386  0.38183
```

```
# Exercise 4
Sigma = matrix(c(1, 0, 0, 0, 2, 0, 0, 0, 3), 3, 3)
pca = princomp(covmat = Sigma)
get_pca_table(pca, Sigma, 3)
```

```
## Importance of components:
##                         Comp.1 Comp.2 Comp.3
## Standard deviation      1.732 1.4142 1.0000
## Proportion of Variance  0.500 0.3333 0.1667
## Cumulative Proportion   0.500 0.8333 1.0000
##
## Loadings:
##      Comp.1 Comp.2 Comp.3
## [1,]                1
## [2,]        1
## [3,] 1
## [1] "The eigenvalues: "
## [1] 3 2 1
## [1] "The correlation coefficients between PCs and variables: "
##      [,1] [,2] [,3]
## [1,]    0    0    1
```

```
## [2,]    0    1    0
## [3,]    1    0    0
```

```
# Exercise 5
X = read.table(file = "outlier3dim.txt", header = T, dec = ",")
Sigma = cov(X)
E = eigen(Sigma)
X_s = scale(X, center = T, scale = F)
Y = X_s %*% E$vectors
colnames(Y) = c("y1hat", "y2hat", "y3hat")
# a
boxplot(Y)
```




```
boxplot.stats(Y[, 1])$out
```

```
## numeric(0)
```

```
boxplot.stats(Y[, 2])$out
```

```
## numeric(0)
```

```
boxplot.stats(Y[, 3])$out
```

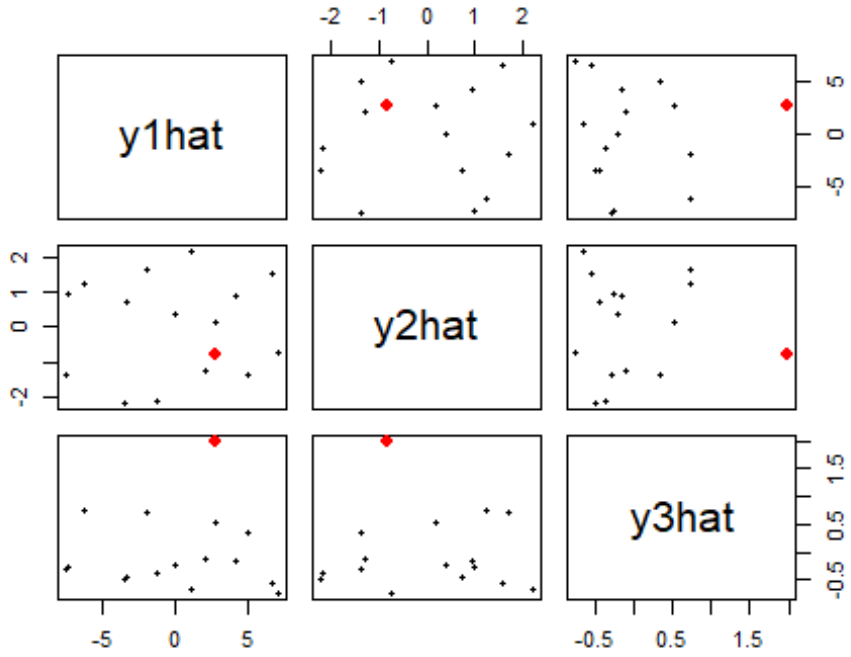
```
## [1] 1.991
```

```
outlier_index = which(Y[, 3] == boxplot.stats(Y[, 3])$out)
```

```
## [1] 10
```

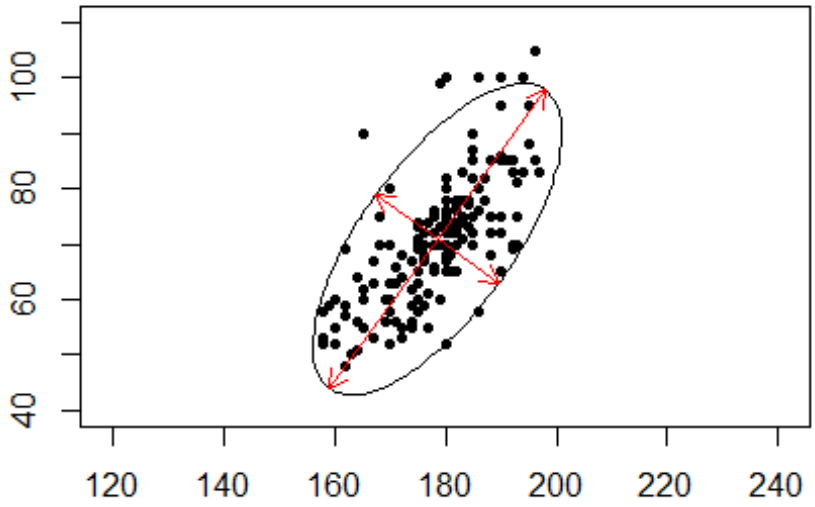
```
# b
```

```
label = rep(1, 16)
label[outlier_index] = 2
pairs(Y, pch = 20, col = label, cex = label)
```

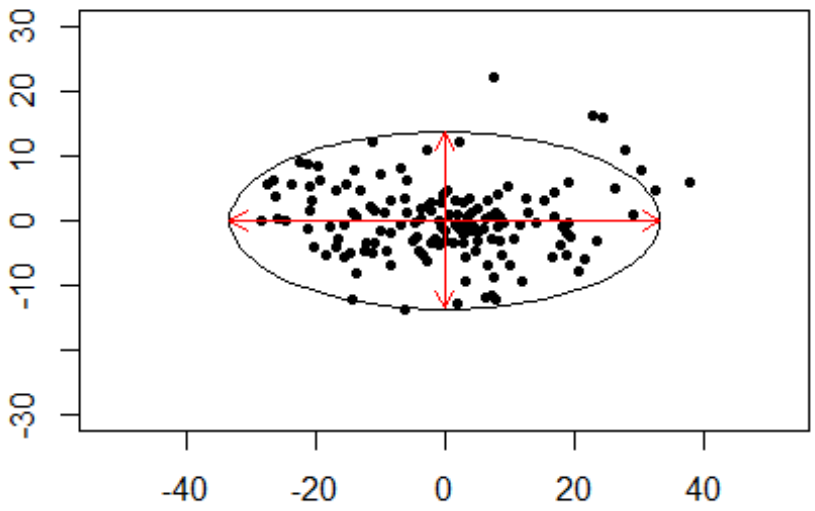



```
# Exercise 6
students2008 = read.table(file = "students2008.txt", header = T, dec =
",")
attach(students2008)
heightweight = data.frame(height, weight)
heightweight = na.omit(heightweight)
detach(students2008)
attach(heightweight)
X = cbind(height, weight)

# a
# pic1

draw_contour_pic = function(X, x1, x2) {
    Sigma = cov(X)
    mu = apply(X, 2, mean)

    f = function(x1, x2) {
        rho12 = (Sigma[1, 2] / sqrt(Sigma[1, 1] * Sigma[2, 2]))
        1 / (2 * pi * sqrt(det(Sigma))) * exp(-1 / (2 * (1 - rho12 ^ 2))
 * ((x1 - mu[1]) ^ 2 / (Sigma[1, 1]) - 2 * rho12 * (x1 - mu[1]) / sqrt
(Sigma[1, 1]) * (x2 - mu[2]) / sqrt(Sigma[2, 2]) + (x2 - mu[2]) ^ 2 /
```

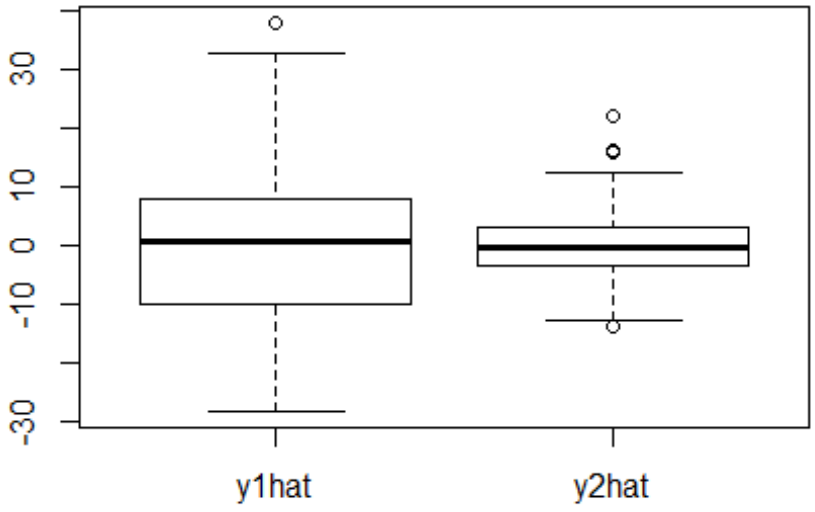
```
(Sigma[2, 2])))
    }
    z = outer(x1, x2, f)
    level = 1 / (2 * pi * sqrt(det(Sigma))) * exp(-0.5 * qchisq(0.95,
2))
    contour(x1, x2, z, levels = level, asp = 1, drawlabels = FALSE)

    points(X[, 1], X[, 2], pch = 20, cex = 1, asp = 1)

    c2 = qchisq(0.95, 2)
    A = solve(Sigma)

    len = c(sqrt(c2 / eigen(A)$values[1]), sqrt(c2 / eigen(A)$values
[2]))

    draw_arrow = function(i, flag) {
        if (flag == 1) {
            arrows(mu[1], mu[2], mu[1] + len[i] * eigen(A)$vectors[1,
i], mu[2] + len[i] * eigen(A)$vectors[2, i], length = 0.1, col = "red",
 asp = 1)
        }
        else {
            arrows(mu[1], mu[2], mu[1] - len[i] * eigen(A)$vectors[1,
i], mu[2] - len[i] * eigen(A)$vectors[2, i], length = 0.1, col = "red",
 asp = 1)
        }
    }

    draw_arrow(1, 1)
    draw_arrow(1, 0)
    draw_arrow(2, 1)
    draw_arrow(2, 0)
}

x1 = seq(140, 220, length = 40)
x2 = seq(40, 110, length = 40)

draw_contour_pic(X, x1, x2)
```



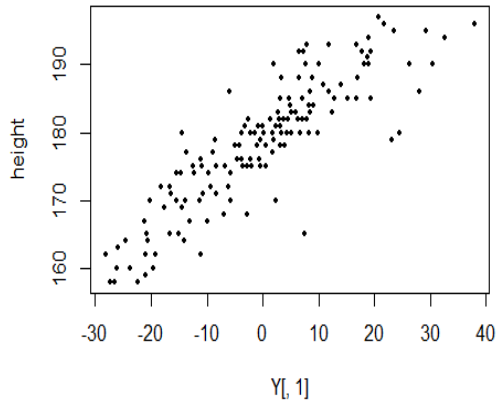

```r
# pic2
Sigma = cov(X)
E = eigen(Sigma)
X_s = scale(X, center = T, scale = F)
Y = X_s %*% E$vectors
colnames(Y) = c("y1hat", "y2hat")

y1 = seq(-40, 40, length = 40)
y2 = seq(-30, 30, length = 40)

draw_contour_pic(Y, y1, y2)
```

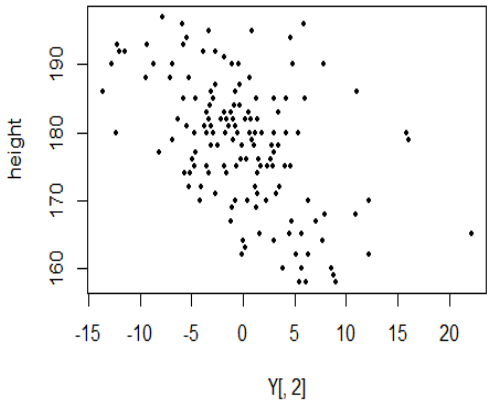


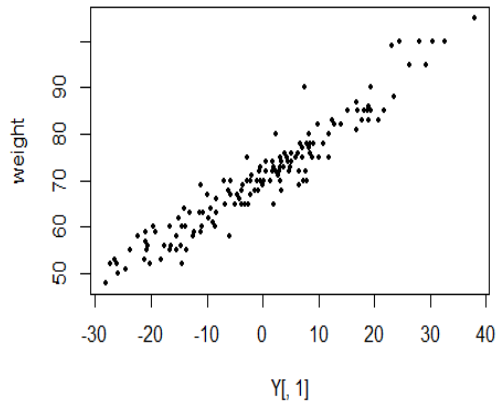

```r
# pic3
boxplot(Y)
```

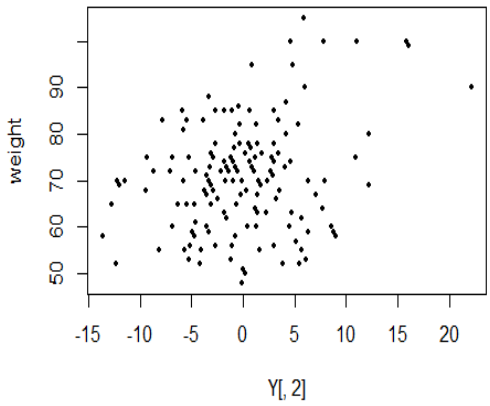


```r
# pic4
plot(Y[, 1], height, pch = 20, cex = 0.8)
plot(Y[, 2], height, pch = 20, cex = 0.8)
plot(Y[, 1], weight, pch = 20, cex = 0.8)
plot(Y[, 2], weight, pch = 20, cex = 0.8)
```



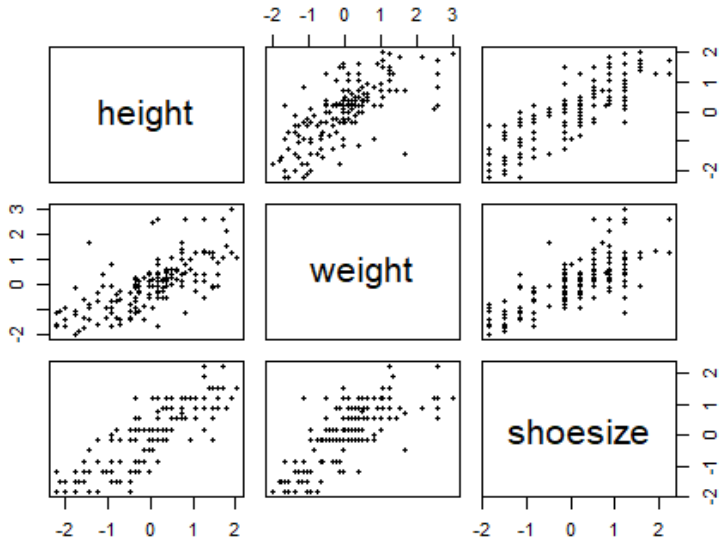




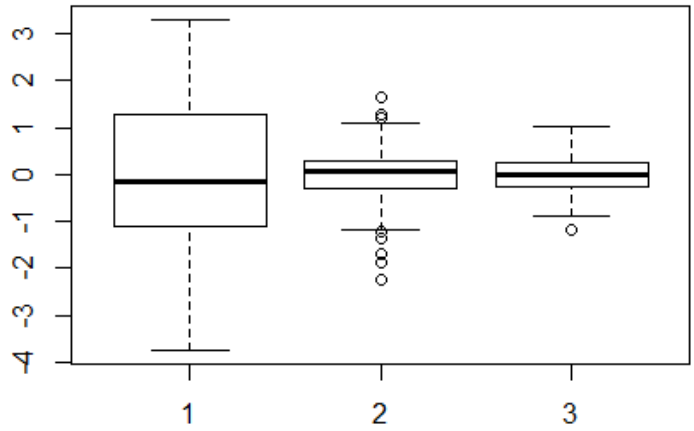



```r
# b
options(digits = 4)

Sigma = cov(X)
pca = princomp(X)
get_pca_table(pca, Sigma, 2)

## Importance of components:
##                            Comp.1 Comp.2
## Standard deviation     13.5484 5.5404
## Proportion of Variance  0.8567 0.1433
```

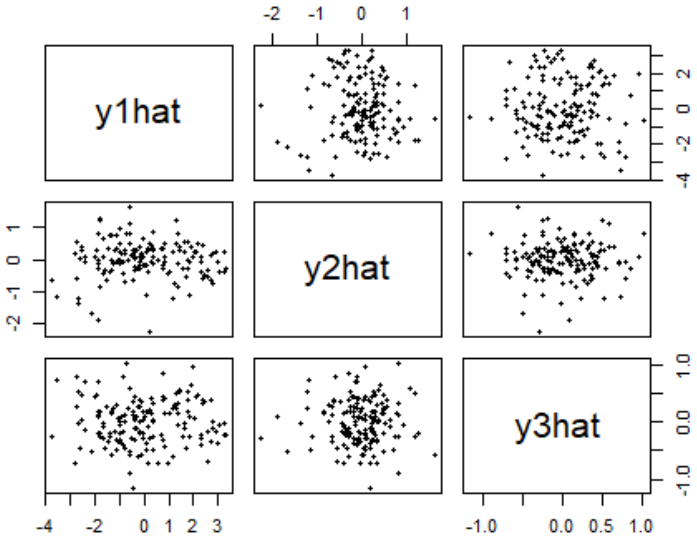
```
## Cumulative Proportion   0.8567 1.0000
##
## Loadings:
##        Comp.1 Comp.2
## height  0.587 -0.809
## weight  0.809  0.587
## [1] "The eigenvalues: "
## [1] 184.72  30.89
## [1] "The correlation coefficients between PCs and variables: "
##        [,1]    [,2]
## [1,] 0.8713 -0.4908
## [2,] 0.9586  0.2846

options(digits = 7) # default
detach(heightweight)

# Exercise 7
students2008 = read.table(file = "students2008.txt", header = T, dec =
",")
attach(students2008)
body = data.frame(height, weight, shoesize)
body = na.omit(body)
detach(students2008)
attach(body)

X = cbind(height, weight, shoesize)
X_s = scale(X, center = T, scale = T)
# pic1
pairs(X_s, pch = 20)
```



```r
# pic2
R = cor(X)
E = eigen(R)
Y = X_s %*% E$vectors
boxplot(Y)
```




```r
# pic3
colnames(Y) = c("y1hat", "y2hat", "y3hat")
pairs(Y, pch = 20)
```



```r
# b
options(digits = 4)
pca = princomp(X_s)
get_pca_table(pca, R, 3)
```

```
## Importance of components:
##                         Comp.1 Comp.2  Comp.3
## Standard deviation     1.5824 0.5702 0.38940
## Proportion of Variance 0.8401 0.1091 0.05087
## Cumulative Proportion  0.8401 0.9491 1.00000
##
## Loadings:
##          Comp.1 Comp.2 Comp.3
## height   -0.582  0.511  0.633
## weight   -0.556 -0.818  0.150
## shoesize -0.594  0.264 -0.760
## [1] "The eigenvalues: "
## [1] 2.5202 0.3272 0.1526
## [1] "The correlation coefficients between PCs and variables: "
##         [,1]    [,2]     [,3]
## [1,] -0.9239  0.2923  0.24715
## [2,] -0.8819 -0.4679  0.05845
## [3,] -0.9429  0.1512 -0.29683
```